THE EVERGREEN PROJECT

LIBRARY SOFTWARE CONFERENCE

CONNECTING EVERGREEN TO EXTERNAL SERVICES

1:05 PM – 2:00 PM EASTERN

MAY 25, 2021

CAPTIONING PROVIDED BY:

CAPTIONACCESS

contact@captionaccess.com

[www.captionaccess.com](http://www.captionaccess.com)

\* \* \* \* \*

This is being provided in a rough-draft format. Communication Access Realtime Translation (CART) is provided in order to facilitate communication accessibility and may not be a totally verbatim record of the proceedings

 \* \* \* \*

>> : We haven't started presenting yet so that might be why you are not hearing anything.

>> ROGAN HAMBY: I could sing to keep people entertained but people would be very unhappy because I'm a horrible singer. I managed to carry a good note once in my life and it was to my shame in a situation that I would never want anyone to report on.

>> : Well, with that it is the top of the hour so if you are ready, Rogan, go ahead and get started. Welcome, everyone. This is going to be Rogan Hamby presenting about connecting Evergreen to external services. I am going to drop the caption link in the chat one more time and I will drop it in a couple more times throughout the event. If you click the link I just put in the chat that is where our live captioner will be recording everything here. Thank you to Marsha for that. And thanks to our sponsors Evergreen Community Development Initiative for sponsoring the platform and Mobius for sponsoring captioning. With that, Rogan, over to you.

>> ROGAN HAMBY: Thank you, Andrea. Let's try this screen sharing thing. Folks should be able to see my screen. If you cannot, please let me know in the chat. My name is Rogan Hamby, I work for the Equinox Open Library Initiative. A lot of you probably know me and know Equinox. For those who don't we are a services provider that provide a resource supporting libraries for things like how Haas subjects plus and Coral and of course Evergreen because that's what we are here for and that's what you're going to talk about.

I'd like to take questions as we go. So if anybody has any, please feel free to drop them in the chat. I will try to keep an eye on the chat, but I will also ask for Andrea to help out. I am talking, Marie, you cannot hear me? Can others hear me?

>> : I can hear you, yes.

>> ROGAN HAMBY: We will keep going. Feel free to ask questions as we go. First thing I want to say is who this is for. The audience of this presentation is not these people. This is not a developer presentation. This is not geared toward highly technical people. Or at least highly technical people who have had time to look into the stuff.

This is oriented toward frontline library support staff, people who are either the most technical person at their library so even though it is not their job they are dealing with support requests related to technology vendors or perhaps they are support service libraries, and, of course, as a system library knows they have a lot of different things to support and they are not going to become an absolute guru and all of the esoterica of anyone product and support. This presentation does have technical elements to it but we are not going to be talking about development, we will not be talking about hard-core server administration. We will be talking about largely things that you can do from a web browser connecting to Evergreen or things that you are going to talk to a vendor about.

I see there is some folks who still cannot hear me. Well, I am going to keep going and hopefully these technical issues will work out. Andrea, can you speak a moment and see if they can hear you?

>> : Sure. Check, check, check. I think most people can hear us. I know these sessions are being recorded and you can also -- we are having live captioning so if you can't hear I recommend clicking on the live captioning link and follow along as well. I'm sorry, we have been having some synchronicity issues with Hopin that we have been putting out as they occur. I apologize for the difficulty.

>> ROGAN HAMBY: Unfortunately, those issues are on Hopin's side so there's nothing we can do but go to the Hopin folks and try to put their attention on them as they come up. It sounds like it's working for more folks now. Some hamster has woken up in its cage. What I was saying very briefly is that this presentation is not for developers, it is not for highly technical work. It is for front-line people who have to talk to technology vendors such as perhaps somebody selling you an external service that validates against your Evergreen patrons or discovery service [indiscernible by captioner] advisory service, any number of things like that. So they need to pull information from Evergreen in some way.

So this presentation is both going to be about pulling information from Evergreen but also about these communication elements, talking to the vendors. What are the kinds of things we are going to talk about? What you can do from the staff client or web browser yourself. What information you can give to a vendor. And when you should escalate these things to your IT staff or hosting provider and how to make that a little bit easier. And we are going to go deep enough into some of the more technical elements so that you know what it is. It's not going to be a tutorial of these things, I'm not going to teach you how to do all of the little bits and pieces, but we will try to give you an overview so that when you talk about some of these methods and pulling information from Evergreen externally you know what they are and what can be done.

Some of the topics we are going to cover. We are going to talk about extracting records from Evergreen. We are going to talk about a service called supercat. Which comes up on the list servers every now and then if you have been a long-time subscriber to the Evergreen list servers. Something called an API. OpenSearch. This thing called web scraping, which I'm going to tell you, you can't see it, but I cringe every time I say it. We are talking about reports. SIP2 and database access. This is a pretty motley crew of ways to connect external sources to Evergreen and get information out. And in most of these cases that is one way through just pulling information out although a couple of these do provide potential for two way information and issue changes to Evergreen.

We will talk about that a little bit. First I'm going to talk about the easy one. Bibliographic record extracts. Now, you can do extracts from the staff client, that is good news. When you want to do that? One, when exports are infrequent. If you only need it one time or you don't know when you are going to need it it is easy to just do it from the staff client. And when it is a fairly low number, by low, I mean, a few thousand. The big limiter here isn't really the size but the amount of time it takes to generate them and send them because if something kills your download method along the way you are going to have to start over from scratch and it's kind of a pain.

And you also need to do it when you can identify a range of bits graphics with IDs. Because the bib extract does require that you supply a list of IDs to bring out and you can do that with reports. You can go to reports unsay I want to list all the ideas of records that were edited last year. For some reason for stop or edited by a certain staff member and I want to put those into an extract. Or when they're in a bucket, you don't have to provide IDs from a report, you can also do it from a bucket. These are all convenient ways to get a list of bib IDs and graph IDs.

But sometimes it makes more sense to go to your hosting provider, your IT staff, however you are running Evergreen, and to request an extract from the server-side. One of this is what it's incurring infrequently. When you are providing bibs, new bibs, to some sort of readers advisory service every month and you know they want them on the first of every month why have somebody do that manually? That can be set up by your IT support staff to run in currently. Also, it's a very large number of records and you might want done from the server. If you have 700,000 records in your database it may not make sense to download them from a client, especially if you have [indiscernible by captioner]

But it can be run very reliably from the server. Also, if the criteria is too complicated for a staff client report. The reporter in Evergreen is very powerful, it does a lot of things but sometimes you can have criteria that are just too complicated for it. And in that case somebody on the server-side may need to put together a SQL report to generate those ideas into the export. What the IT staff will need to know if you do ask them to do a bibliographic export for them is what you send the output to.

This is not going to be an email address. They are going to need something like an FTP address. Because the files are going to be probably too large for email unless we are literally talking about dozens. Are we talking about all records or do you need to supply some criteria? If there is criteria be as specific as you possibly can. Do you need the records in MARC 21 or MARC XML? If you don't know what those are probably MARC 21, that is kind of the standard, the most standard form of moving MARC records around.

Are we going to include holdings in this? Do you just want the bib itself or do you want tags with holdings? Now, this can be important because MARC 21 records have a limit on them. And if the vendor you are sending records to uses MARC 21 it is actually pretty easy if your records have a lot of holdings to break the size limit of MARC 21 records in which case they either need to be able to ignore the size limits of MARC 21 or they need the records in MARC XML. But with these criteria it is pretty easy to talk to support staff and get a bib extract set up.

Supercat. Supercat is not something that comes up every day on the service but if you have been around you have probably at least heard of it. And this is a link to the Evergreen docs that discuss it. It goes more in-depth then I am going to hear but let me give you a brief introduction. Supercat is another way of getting data about holdings out of Evergreen to an external service. Again, this could be discovery layer, it could be readers advisory service, it could be any number of things. It could be an app that is been developed on [indiscernible by captioner] for helping people browse your collection.

And what supercat does is through URLs, HDP web browser type communication, get information on bibs, ISBN's, and meta records. Let's go ahead and go through an example. I am not going to be brave and do a live demo and here. I'm just going to do stuff on slides. But these slides will be posted later on the conference website and the video recordings will be available later on YouTube. If you take this URL and you replace your host with your actual Evergreen IIS URL then you can get this. This huge blob of text is the different kinds of formats available from supercat.

This may sound really cryptic right now but when we talk about these formats these are the formats supported by URL Evergreen. Now, it may not match this list exactly, it may depend on which version of Evergreen you're running, but that hasn't changed recently. I am watching chat and I see people are having audio issues. Are people not seeing these slides change or is that just some folks having issues?

>> : I think that's only some folks having issues. I can see we are on your supercat slide with the blob of text. Those of you having issues maybe you can try changing browsers. I'm really sorry for the amount of pop in issues I'm having. I'm writing them all down and documenting to take to Hopin later today. I don't know what to tell you, I am sorry.

>> ROGAN HAMBY: Fortunately, the video will be available on YouTube afterwards. I notice somebody saying they're having issues on a Mac, I'm watching this as a viewer on Chrome Mac so there may be plug-in issues or something else. I don't know. I have to move ahead for now and use captions best you can. These are the formats and some of these formats are things like OPAC, which is only going to be really meaningful to Evergreen HTML holdings but some of these like the atom formats and the RSS formats are very widely used outside of the library community.

So vendors that are looking to set up an external service that pull information from Evergreen should be familiar with how these types of data feeds look. They will probably have to do some development on there and to make use of the specific information being sent but they know what atom and RSS and things like that are. Let's look at this in practice. I should mention in this example I am pulling bib records which is indicated by this key term record at the end of the link. You can also pull authority and meta-records by just changing that. So instead of formats/record changing the formats for /authority or meta-record and you can get a list of all of the formats supported by those.

Now we will show a natural retrieval example. This is similar to before except now we are going to say retrieve, we are going to list one of those formats that we got from that previous listing but I had a record type any record ID. Again, I am not going into the detail to show you how to do this yourself all of the time but I want you to know that this exists and what is possible when you talk to vendors. Here is an example, one of my test servers. And small dot lib was my test server. Asking for a bib record in the atom format and I'm giving it a then record. This is what it gave me.

It says atom X LL feed that gives me the various information about the record including the title, author, copy of the Hobbit and so on and so forth. This is a way that external vendors can pull information from Evergreen. Now, they do need the bib record IDs for this. I'm going to show you another example of this. This time instead of the atom XML I'm going to do HTML holdings. This is probably for a human being evaluating this easier to read and more useful and it certainly can be used in a variety of circumstances.

You can use this to embed inside of an HTML page. But it is probably less useful to vendors who are going to want structure to data to parse into their own products. Now, I mentioned this post a single bib record by its ID. Obviously, the vendors going to need a list of IDs from somewhere. They are not going to call you up on the phone and say can you give me a list of all of your group ID since last time we did this? That would be pretty inefficient. There's going to be some sort of automated way to do that.

You could have somebody sit there and run a report every week and email it to them but that is not terribly efficient either. The focus of having computers is to do things automatically because we don't want to, right? So that is why supercat has another tool. It's called fresh meat and it is to provide a feed that will give you a list of bib IDs that you can limit by date as having either been imported or edited up to a certain number. So in this particular case I can say I want to go to the server, I want an atom format, bib records that have been imported up to 100 of them since a certain date.

And this is an example of that. Again, my test server ends with atom bib I want everything edited up to 100 of them since the 1st of January 2020. Then we start getting the support. This time instead of asking for a bib ID and getting a single record I'm going to get up to 100. And if you have some be coming to you and saying hey, we have this service to display information about new books on this community web thing but we need stuff from you to help display it this is all they need. This would give them that data. And let them pull that information. Without you having to do anything special.

Now, when you want to use supercat? Well, the short answer is anytime a vendor needs a real-time update of records. Because that is an important thing here. Sure, a lot of this you can pull from reports, you can write a report to say I want the titles and the authors and all of that stuff of the records edited since a certain date and you can generate that periodically and send it off but this is real-time. This will reflect constantly. Somebody could create a supercat feed and programmatically say every hour give me all of the new records added in the last hour. And a good example of something that this would be very useful for is discovery layers.

Now, let's talk about practicality points. In most cases they probably want to start with a baseline of information. You're probably not going to want to tell somebody if they are doing a discovery layer which needs a listing of all of the bibs you have, they are not going to want to try to pull every record in your system with supercat. You are probably either with a report or bib extract going to offer to give them a baseline and they will use a tool like supercat to get new changes. Now, you will probably want to reconcile periodically that this is true of any time you are doing this sort of external data. How do I set it up? The short answer is you don't. It's already running on your server. It's an inherent part of Evergreen. Part of the point of this presentation is you don't have to be the developer for this. If somebody is coming to and selling a product as a library you point them to the docs and tell them this is a tool in Evergreen for you to do this.

That is on them. Now let's talk about unAPI. Or as I tend to call in my head unAPI, which is wrong. That is what I can do. If I say to unAPI, I mean, unAPI. Again, thanks to our documentation team we have good docs on API. It's essentially an alternative to supercat. Like supercat, you'll pull information from URLs but it's called the unAPI because it's even though it's not unAPI it has some API like functions. UnAPI is basically a programming interface, a way to externally say give me this stuff and it will send it back. Supercat is too simplistic to be considered unAPI. UnAPI is still too simple as it it's closer to [indiscernible by captioner]

Here is an example. Because a little bit more complicated, a little bit more intimidating but again, the take away I want you to have from this is what is possible with it, not how to do it. If you're talking to an external vendor and they are looking at the writing support you point them to the docs and then figuring out the mechanics is up to them. In this example we have a URL again, plug in your host and down here you are going to say that we want bibliographic record entry records, we are going to provide a bib ID of one or more classes and a format.

You can leave the format off to see the available ones, just like we did in supercat. In this case it's a much smaller list. Holdings XML file, MARC XML, and mods 32. Any of these three are probably useful to external vendors. Mods 32 is going to be more library specific than the other two but if they are a library vendor offering you a service they may be familiar with it. They certainly should know how to parse holdings and MARC both XML format.

Let's talk about classes. This is a longer list. This is the kind of stuff you can send back to you. BRE is the simplest, that is a bib record. Then we have bib they are record entry feed, that is a list of bib records. Call numbers, all kinds of stuff. And like before we are going to do an example and give you can -- some context. Here we are saying I'm asking for a bib record information, I'm sending you a bib record ID and I want holdings XML and call number data returned in the format of mods 32.

This is what it sends. Very simple, it's very straightforward and it gives you that mods 32 data. You can see in the library the call number, because we requested it, and to the copy level data and the holdings. Now, let's expand that a little bit. This is the same example as previously except I have added ACP into that list. Right here. The holdings XML, call number, ACN, and ACP is copy. The asset copy. Now we are going to see output similar to before but now we are also going to see some specific copy level information like barcode, Serc modifier book, OPAC visible, that sort of stuff.

What makes an unAPI better than supercat? Both give us bib data, what makes it really better? It supports limbs and offsets. That's not something you need to worry about too much but the short version is let's say a service does need to pull a thousand records and that's really too many people at one time. Well, with supercat it's all or nothing. With unAPI you can say give me the first 100. And then the next hundred and the next and you can split up that record for so many writing something to pull that information from Evergreen.

You can specify an org unit and org unit depth. This is really important for the holdings information, especially in a consortium because it means that you can pull information specific to a library even if many different libraries have all of their holdings on one record. For an example where one library consortium is contracting out for an app but it's not eight consortium contract just for that library, this becomes really useful for that vendor. And you can retreat by item barcode instead of bib, which is really nice. So when would one of your vendors want you to use unAPI? Basically, when they are talking about stuff to display and copy level information, custom app on the same sort of things that you do supercat.

Until a few years ago I would have told people you might want to design a carousel using this. That is a good example of something that you can use this for. Of course, carousels are now made and supported in Evergreen and there is no need. But if you wanted to reinvent the wheel. Let's talk about OpenSearch. Again, many thanks to documentation community, they have provided us an invaluable tool by giving us information but I'm actually going to quote Wikipedia here. They give a description of what open searches. It's a collection of technologies that allow the publishing of search results in a format suitable for syndication and aggregation. It goes back a good ways now, 16 years. It never became as widely used as I think it should have. But Evergreen doesn't support it, which is really nice.

It's not to be confused with the fork of elastic search by Amazon which also called [indiscernible by captioner] search. Totally different things. Again, what would you say to somebody about this? When would you invoke OpenSearch? Again, when you're talking to a vendor and they are saying how do we get this information from your ILS to use on our product? You can say one of the things Evergreen support is OpenSearch and if they say great, our product supports OpenSearch to for pulling information then you can just point them to the stocks and they are good to go. You're done. They might need some things like your org name and whatnot but you are halfway there.

Web scraping. Web scraping for those who don't know is basically the practice of pointing a program that acts like a browser and reads web pages off of a server and pulls the information off. The short answer is don't do it. And if you have a vendor that says don't worry about it, we will scrape your website, please ask them not to. It is not reliable, it is messy, there's just no good reason to. All of these other things we have talked about, supercat unAPI, it will give them structured information as reliable and clean. This is messy and horrible. And when things happen like you switch -- some libraries are going to be looking to switch to the bootstrap OPAC it will break web scraping for a lot of applications using it. That stuff is going to be presented differently.

Routinely OPAC changes could potentially do this. So why do some vendor suggest it? They see it as simple, they don't have to talk to anybody else. They just handle everything in-house. And if you find yourself in a situation where you do intend to have a vendor use web scraping of your OPAC do talk to your ILS provider first. Whether this is in-house IT people or you're hosted by a next-door service provider like Equinox, talk to them about it first. Because they are probably going to ask you to do some things like not during active hours. Nobody wants a bot doing 10 calls a second for hours and hours at the same time that they are trained to get books checked out to the after school program that's just let out and things like that or during the lunch rush. Just not very [indiscernible by captioner] and limit concurrent hits.

I have seen vendors do web scraping who go why not 150 connections a second? Let's bring that down. Again, what number should they use? Talk to provider first. Talk to your IT provider. They can help you out. And Jason Stevenson made a good point in chat, web scraping is more of a load on your server. And the short answer as to why is when we are presenting to information to humans we process [indiscernible by captioner] supercat and unAPI that structured data, which again, is better for the vendors who need that information it's already in computer form.

It is easy to server up to them. Better ways, web scraping is bad, we don't like web scraping. An initial list of the alternatives, if they give an initial list of bib IDs with data, whether it's from a bib extract or a report and it depends on the service on what they need, you can get that, let them get their database up and running, and then get updates from supercat or unAPI, a discovery service is a good example of this.

You still want to limit the concurrent hits. If they insist on doing web scraping. We are getting down to about the last 18 minutes and I want to leave a little bit of time at the end for discussion and Q&A. I'm going to blast through the last couple points. One is reports. We mentioned before that he can use for bib IDs but honestly, you can pull anything from reports. If you are working with a vendor that just does not know how to use bib records and you want to pull information from 500's, for awards, for their app they are doing for you, you can do that with reports.

You can pull every individual MARC record tag and structure exactly what they want. That's fine. But after this point we are going to start talking about things that are scared and that's because up to this point I have really limited what I have talked about two things that are copy and bib data. We're not totally concerned about those, those aren't sensitive. Nobody cares if the whole wide world knows that they have copies of a particular book. That is not sensitive data. But patron data is sensitive. So we have to be a lot more careful about it.

And this is also where we start getting into talking about -- this presentation is about communication as well as technology. So reports. We talked about you can pull from reports bib and copy data but you can also pull everything else. Now, there are advantages like, notification sent, they can be CSV or XLS and you can pull everything. The disadvantages. Content can't be sent. In the case of the patron reports that's an advantage. There are somethings reports want to. And there's no support for alternative formats. There's more that can be done on the back end on the server although it may not be a quick thing. I will go and tell you when you approach your IT staff or your hosting support staff about alternative forms of reports this is going to have to be a dialogue.

Because what you want and what an external vendor tells you it's easy may not in reality be easy. For a number of reasons. Remember, that external vendor just wants to sell you a contract. They will be very happy to make the problem your support staff problems after the contract is signed. So before you finalize on a product do talk to your support staff about it, before you follow-up. And have a discussion about things like the patron privacy issue. I have seen vendors say yeah, just set up a report to run all the patron's names and emails and phone numbers and just FTP it over to us. No. Whether or not you want to use an external source and send them patron information is a whole other question and it's not a question to be decided lightly. If you sell my preconference yesterday and Becky's keynote this morning about that, but if you do make a decision to do that make sure it is transferred securely, make sure you have agreements with that vendor, that you know how long they are quick to retain the data and who has access to it and all of those kind of things.

I already talked about these base points. Server site reports you can make much fancier. SIP2 is another way that you can get information on the system and you get information about patrons aside from the ports. Now, SIP2 should not be confused with telephony protocol. This is one that was originally specked out by three M and there's a PDF out there that you can grab. I don't call it a stander, I called a specification because it's more of a gentlemen's agreement that is frequently abused than a standard. There had been a number of extensions of SIP2 over the years, many of them are very well and widely supported. But there is no governing body saying it is part of a standard. In therapy parts of it that include language that has been entered a different lead by different entities. In fact, we have some options in this into server that Evergreen uses to which its behavior based on some of those [indiscernible by captioner]

But it is kind of the common language of the library world. Everybody seems to support SIP2. We could have a very long discussion about why other things should have taken it over by now but that is the situation we are in. And it supports both information and transactions. Unlike anything else we have talked about so far up and through the ports and will provide patron information and transactions and by transactions, I mean, you can have SIP2 send back messages and say try to check this book out, try to place the sold. Try to do something with this font, those kinds of things. But the major problem with it is it's not encrypted unless it is. SIP2 itself is plaintext which is not a good thing.

There are ways to tunnel through encrypted tunnels and I do strongly recommend to people that they look at that as an option. And depending on whatever laws governing at your municipality, county, state, or even national level unless you're in the U.S., U.S. doesn't have any natural laws regarding this, they may require you to tunnel it unless you fall under an exception. There is no hard and pastoral about her but in that regard. Too much variation in the world [indiscernible by captioner]. Now, I do want to spend a second demystifying SIP2. I work with libraries with front-line staff all of the time who do not have really dedicated IT staff, meaning they hire external people and they end up talking to a vendor and then they talk to me as one of the hosting provider people and they are disco between talking about sip and it's very cryptic to them.

It's very simple, really. SIP2 is a protocol. And all the protocol is it's a series of rules for communication. Just like in the noncomputer world. When you hear about it protocol for steak dinner it is rules for how people communicate at a steak dinner. SIP2 particles on the same things, rules on how can beers talk to each other. I'm not going to get into the technical details about message response personnel if that but all sip does is the one thing is send a message to another and get a response. I can even book vendor saying I'm setting this barcode and password are they legit? And Evergreen says yes or no. That is pretty much it. Don't fill intimidated by this kind of talk.

You signed up for a new e-book service and then you use a sip account, there's nothing magical going on and I will go and say this, if they say Evergreen won't let us check this book out there may be an issue that needs to be looked at in regard to messages and touts conforming to what they expect but Evergreen certainly can't tell any book vendor whether or not to check the book out. That decision is happening on their side. We are just sending information back. There may be an issue to look at though. What do you do when you're having a SIP2 account made? When you're talking to your ILS help desk, whether it's internal IT staff or service provider you want to give some information.

We will say this is an e-book provider. Provide the IP address the SIP2 request is coming from. If you white list traffic then your IT staff are going to need that. If you don't like links to traffic you can ignore it but better to be prepared. The name of the account. If the e-book vendor wants to log in with an account we are awesome and that needs to be provided to your staff so they can set up that account with that name. If they don't care we can make up a name.

And then the IT staff will have to go in and do two things, really three things. Create an Evergreen account, a sip server account, these are separate but they must have the same username and password and then restart your SIP2 service. For that to take effect. Then in exchange with the ILS support staff will give you is a vendor inversion, this is what say the e-book provider usually expects, the vendor in our case is going to be Evergreen because much of the library world still does not knowledge the idea of an ILS that is not controlled by corporate entity. The version Evergreen you are running. The sip server address and port, which are IT ILS hosting staff can provide you. And then that username and password.

Then an institutional lake patient cut. That will be all that your e-book vendor needs to set up your subaccount. Now, the less when I will bring up his database access. Again, this is a way to pull up patron information. Obviously, which reports their concerns about patron information. It should not be sent by email. It should not be sent unencrypted. SIP2, because that's patron information I recommend encrypted tunneling. Database access takes all of those concerns and adds more on top of them. And it feels like I have seen a spike in vendors asking for direct database access recently.

This has major privacy concerns both because it takes what could be a stream of information and turns it into a fire hose. And if they have wide access that could potentially change patron information. First, check your laws governing patron information. That really applies to reports with patron information and sip as well. And think about what you have to do to cover your own posteriors. I don't know how to say this too strongly, bring lawyers in. I am not a fan of bringing lawyers and for every little thing but if you're going to provide someday database access to your system then I would strongly recommend make sure they really need it and that you really want whatever this product agreement is.

You need to have legal waivers in place. And if they look at you and say we are not going to sign a waiver my personal recommendation would be find a different product. To fill whatever this need is. And I suspect that whatever your funding institutions, governing institutions are, whatever your library is, will want waivers in place for database access. And someone asked a couple comments and chat, I will just work quickly. Which are asking for database access, I'm not going to get into that that something that will be on YouTube but there will be other people responding with some that may or may not include the ones I'm thinking a.

What exactly would a waiver be waiting? A waiver needs to include several things. One, if they only have read access it essentially needs to waive and it needs to address liability for if they do something with that information they shouldn't. Enter contracts with an entity that has patron information need to address a number of points including how they retain it, how they get rid of it, who has access to it, how it's used, all of those things. And also if they have any kind of update access to your database and I'm not showing my camera so you can't see me cringing in pain as I say that, the waiver needs to waive you of liability for what they do.

If they hose your entire database, yeah, I need to take off -- I work for a hosting provider hat for a second and put on my I worked in public libraries for 20 years hat and say I would just tell somebody no. If they ask for API access. Just no. But are there people out there that want to? Absolutely. We have a fume has left some blessing this last one. This is for those frontline folks out there who are not techies but talk to vendors all the time. This is going to be a phrase you here. Hey, can we have API access? API is a term that technically means one thing but it is used in a lot of other ways.

Does Evergreen have API's? Yes. Does Evergreen have publicly web accessible restful APIs in the sense that they mean when they ask that? No. And part of that is the privacy of patron information and all of this other stuff. Jason said Evergreen is basically unAPI. Yeah. Just not in the way a lot of entities in the wider world tend to say it. And part of it is intentional decisions to provide barriers that protect information. When they come to and they say can we have API access if they are just looking for a bib and Eitan data take them back to supercat and unAPI. They can pull what they need in those ways. If they need patron information that is a whole other discussion.

That brings me to our end. We have about three minutes, any question or discussion points people want to have?

>> : I did not notice any additional questions other than the ones that you grabbed.

>> ROGAN HAMBY: I hope this was useful to people. Again, it is not a super technical presentation. It was not meant to be. You can have a lot of fun actually digging into things like supercat and unAPI. Sometimes people ask me I'm not a developer, I don't have test servers, what can I do? To look at some of the stuff? Well, that. If you want to play with Coral or Python and look at ways to programmatically pull holdings information from Evergreen, you can use these. Absolutely.

>> : Thank you so much, Rogan. That was very interesting and a really nice follow-up to some of your points with what Becky talked about in her keynote. I apologize for those of you who are having audio and other difficulties. I had been taking notes of the issues with Hopin and will be followed up with them and hopefully this will settle out. And remember your key browser troubleshooting issues. Try a different browser, hard refresh. All of those things that we all talked about in the context of the web client as well.

>> ROGAN HAMBY: I will say were quickly, I have also antidotally heard from a couple people that some privacy plug-ins may be causing problems. Probably because Hopin wants to track your existence across tabs. You may want to consider disabling some plug-ins if you're having issues

>> : Hopin's own article about that recommended doing incognito or private rows as a way to skirt that as well as any potential caching issues. We will figure out, we are flying this plane -- we are building this plane what we are flying it. Thank you, Rogan. I will go ahead and release you all. We have ten minutes until our next session which will be coming up in this track. It will be picture-perfect, Icons in Evergreen. That would be with Kate Coleman and Jennifer Weston.

We will see you all in a few minutes.

[ end of meeting ]